

(19) World Intellectual Property Organization
International Bureau



(43) International Publication Date
31 January 2002 (31.01.2002)

PCT

(10) International Publication Number
WO 02/08839 A1

(51) International Patent Classification⁷: **G05B 17/02**

(21) International Application Number: **PCT/EP01/07235**

(22) International Filing Date: **27 June 2001 (27.06.2001)**

(25) Filing Language: **English**

(26) Publication Language: **English**

(30) Priority Data:
00306250.2 21 July 2000 (21.07.2000) **EP**

(71) Applicant (for AE, AL, AM, AT, AZ, BA, BG, BR, BY, CH, CN, CO, CR, CU, CZ, DE, DK, DM, DZ, EC, EE, ES, FI, GE, HR, HU, ID, IS, JP, KG, KP, KR, KZ, LC, LR, LT, LU, LV, MA, MD, MG, MK, MX, NO, PL, PT, RO, RU, SE, SI, SK, TJ, TM, TR, UA, UZ, VN, YU only): **UNILEVER N.V.** [NL/NL]; Weena 455, NL-3013 AL Rotterdam (NL).

(71) Applicant (for AG, AU, BB, BZ, CA, GB, GD, GH, GM, IL, KE, LK, LS, MN, MW, MZ, NZ, SD, SG, SL, TT, TZ, UG, ZA, ZW only): **UNILEVER PLC** [GB/GB]; Unilever House, Blackfriars, London, Greater London EC4P 4BQ (GB).

(71) Applicant (for IN only): **HINDUSTAN LEVER LTD** [IN/IN]; Hindustan Lever House, 165-166 Backbay Reclamation, 400 020 Mumbai (IN).

(72) Inventors: **KLAFFKE, Werner**; Unilever Research Vlaardingen, Olivier van Noortlaan 120, NL-3133 AT Vlaardingen (NL). **PATEL, Shail**; Unilever Research Port Sunlight, Quarry Road East, Wirral, Wirral, Merseyside CH63 3JW (GB). **RABONE, Jeremy, Andrew, Leslie**;

Unilever Research Port Sunlight, Quarry Road East, Wirral, Merseyside CH63 3JW (GB). **RUSSELL, Stephen, William**; Unilever Research Vlaardingen, Olivier van Noortlaan 120, NL-3133 AT Vlaardingen (NL). **TISSSEN, Johannes, Thedorus**; Unilever Research Vlaardingen, Olivier van Noortlaan 120, NL-3133 AT Vlaardingen (NL).

(74) Agent: **WURFBAIN, Gilles, L**; UNILEVER N.V., Patent Department, Olivier van Noortlaan 120, NL-3133 AT Vlaardingen (NL).

(81) Designated States (national): AE, AG, AL, AM, AT, AU, AZ, BA, BB, BG, BR, BY, BZ, CA, CH, CN, CO, CR, CU, CZ, DE, DK, DM, DZ, EC, EE, ES, FI, GB, GD, GE, GH, GM, HR, HU, ID, IL, IN, IS, JP, KE, KG, KP, KR, KZ, LC, LK, LR, LS, LT, LU, LV, MA, MD, MG, MK, MN, MW, MX, MZ, NO, NZ, PL, PT, RO, RU, SD, SE, SG, SI, SK, SL, TJ, TM, TR, TT, TZ, UA, UG, UZ, VN, YU, ZA, ZW.

(84) Designated States (regional): ARIPO patent (GH, GM, KE, LS, MW, MZ, SD, SL, SZ, TZ, UG, ZW), Eurasian patent (AM, AZ, BY, KG, KZ, MD, RU, TJ, TM), European patent (AT, BE, CH, CY, DE, DK, ES, FI, FR, GB, GR, IE, IT, LU, MC, NL, PT, SE, TR), OAPI patent (BF, BJ, CF, CG, CI, CM, GA, GN, GW, ML, MR, NE, SN, TD, TG).

Published:

- with international search report
- entirely in electronic form (except for this front page) and available upon request from the International Bureau

For two-letter codes and other abbreviations, refer to the "Guidance Notes on Codes and Abbreviations" appearing at the beginning of each regular issue of the PCT Gazette.

(54) Title: **METHOD FOR SIMULATING CHEMICAL REACTIONS**

(57) Abstract: A process for simulating complex chemical reaction pathways, wherein the simulation is based on transformations with relative probabilities that helps predicting the outcome of processes that may involve multiple chain reactions and/or parallelism and/or feedback or feed forward loops.



WO 02/08839 A1

METHOD FOR SIMULATING CHEMICAL REACTIONS

Field of the invention

5 The present invention relates to a process for simulating (chemical) reactions. More in particular, this invention relates to a simulation of complex chemical reaction pathways, wherein the simulation is based on reactions with relative probabilities.

10 Background of the invention

Simulating chemical reactions is a useful tool in a wide range of industries, and applications are e.g. designing the most efficient reaction pathways, risk analysis in chemical plants, formation of flavouring or aroma compounds, biochemical pathways,
15 processes of sulphonation and others.

There are a number of approaches in the literature which simulate reaction pathways either synthetically or retro-synthetically. These may be summarised as:

- (i) Search engines based on large databases, e.g. CASREACT, CRDS,
20 BEILSTEIN, ORAC, REACCS, SYNLIB, and CHEMINFORM which classify reactions and allow searches by molecule fragments and functional groups.
- (ii) Computer-aided Synthesis, e.g. PSYCHO, DARC-SYNOPSIS and REACTION simulates reactions in the forward direction from start reactants.
- (iii) Computer-aided Retro-synthesis e.g. LHASA, RETROSYN, OCSS and
25 SYNCHEM, builds the synthetic tree for a user-specified molecule. Some also support synthesis in the forward direction, i.e. allow the user to specify start compounds to predict end products e.g. SOS^[4], MARS^[5] and SYNGEN.
- (iv) Mathematical models, e.g. energy calculations (EROS) or electron density calculations (CAMEO), are used to predict chemical reactions.
- 30 (v) Combinatorial Chemistry e.g., Diversity Explorer ^[1], Chem-X ^[2], or Legion ^[3], for building virtual combinatorial libraries.

Bador^[6] et al. give a review of the approaches listed under (i) to (iv).

As the intended use of these approaches is generally an aid for the synthetic chemist, they have drawbacks such as: user input is required to proceed, and/or only a single
5 branch of the reaction pathways is followed, or other disadvantages. These disadvantages are particularly a handicap when wishing to model complex chemical reactions that have for example reactions or transformations that occur subsequently, and/or in loops (forward, backward, or mixed), and/or in parallel.

10 In order to predict the outcome of processes that involve multiple chain reactions, a system that can cope with inherent parallelism and feedback or feed forward loops, and operate without user interaction to construct the complete reaction graph, is preferred. Prickett and Mavrovouniotis^[7] have developed a theoretical system that models generic complex reaction systems. This iteratively applies known elemental reaction steps,
15 according to theoretical chemistry, to the reactants and all intermediates.

This method has some disadvantages such as:

- it is theoretically sound, but may not take into account the practical difficulties with scaling up a theoretical approach for industrial purposes,
- 20 - it does not take into account the different rate constants or kinetics of the reactions involved,
- it does not describe a way of validating the results, and updating the simulation using experimental data.

25 Summary of the invention

Hence, there was a need for a method for modelling or simulating (complex) chemical reactions or processes that helps predicting the outcome of processes that may involve multiple chain reactions, a system that can cope with inherent parallelism and feedback
30 or feed forward loops, and operate without user interaction.

It has now been found that the above may be achieved (at least in part) by a method for simulating a chemical process, which process may comprise multiple branches of reaction pathways and/or feed back/forward loops and/or parallel reaction branches by an iterative procedure of applying:

- 5 - a 'Reaction Set' describing transformations and their probabilities that may take place in the chemical reaction or process on
- a 'Soup' of molecules representing the state of the system.

Detailed description of the invention

- 10 The system according to the present invention is similar to the system of Prickett and Mavrovouniotis^[7], but better in three significant ways:
- 1) taking into account reaction rate constants as reaction probabilities
 - 2) and optionally heuristic blocking of the reactions into subsets that guide the reactions in a computationally effective manner
 - 15 3) and optionally fine-tuning the reaction and reaction rate databases by comparison with experimental results.

The simulation of complex chemical reaction pathways according to the present invention (hereafter called Iterated Reaction Graphs - IRG) model complex reaction

20 pathways by simulating the reaction steps in parallel. An Iterated Reaction Graph has two main elements:

1. A 'Soup' of molecules representing the current state of the system
2. A 'Reaction Set' describing transformations (= simulated reactions) that may take place in the chemical process that is to be modelled or simulated, and probabilities
- 25 (= simulated reaction rates) of said reactions to yield molecules.

ad 1) In the 'Soup', molecules may be represented by any computer readable format, e.g. expressed as SMILES^[8], a simple line notation of 2-dimensional connection tables.

- 30 Preferably, during the iterative procedure the newly formed compounds are added back to the Soup, which forms (part of) the virtual mass distribution. Additionally, it is

preferred that the Soup at the start of the simulation is equal to the starting mixture of molecules.

ad 2) In order to describe the reactions that may take place in the process that is to be simulated the 'Reaction Set' may suitably contain (in computer readable format):

- a reaction database, which contains various transformations that may take place in the reaction or process to be simulated. These transformations can usually be found in literature.
- a reaction kinetic database, containing probabilities for transformations to take place in the reaction database, simulating kinetic data such as rate constants for the reactions.

Furthermore, the IRG contains a computer programme directly loadable in the internal memory of a computer, comprising instructions for the simulation of complex chemical reaction pathways by iteratively applying a set of operations or computer instructions to:

- A 'Soup' of molecules representing the current state of the system
 - A 'Reaction Set' describing transformations and probabilities that may take place in the chemical process to be simulated
- to produce molecules, for simulating complex chemical reactions when such product is run on a computer, and wherein the computer programme contains two main elements:
- a) computer instructions for applying the transformations using the reaction set described above,
 - b) computer instructions for the iterative procedure of selecting molecules, applying the transformations and producing output.

The computer programme also contains typical components such as a user interface, methods of inputting and editing data, methods of probing the progress, methods for outputting results and so on.

The IRG is the iterative application of a 'reaction set' which is applied on a 'soup' of molecules. The iterations are over all reactions, and over all candidate molecules, in

the various reaction blocks. Preferably, the iterative procedure is coded as a computer programme directly loadable in the internal memory of a computer

The invention further comprises a computer program product directly loadable into the internal memory of a digital computer, comprising software code portions for the simulation of complex chemical reaction pathways by iteratively applying a set of operations or computer instructions to:

- A 'Soup' of molecules representing the current state of the system,
 - A 'Reaction Set' describing transformations that may take place in the chemical process to be simulated, with their respective probabilities,
- to produce molecules, and wherein the iterative procedure is coded as a computer programme directly loadable in the internal memory of a computer, wherein the iteration is coded as a computer programme, for simulating complex chemical reactions when such product is run on a computer.

Each reaction may be coded as a computer program that takes connection table input (reactants), carries out necessary rearrangements (reactions), and produces a connection table output (products). In the present document such coded (or virtual) reaction is called 'transformation'.

At a simplistic level the reaction base operates on the molecular soup to form products:

Reaction Set: Molecular Soup -> Products

The full complexity of the possible reactions may be modelled by iterating through this 'equation', feeding the products back into the Molecular Soup and running through the Reaction Set again, which is a part of the IRG (Figure 1).

The full reaction graph^[8-12], where molecules are nodes and reactions are arcs may be defined as the set of triplets:

{<Substrate> <Reaction> <Product>}

For example the text below is a small fragment of a Reaction Graph, containing 3 triplets (molecules coded in SMILES):

```

      C(=O)C(C(=CC(=C)O)O)O   R1_1_6_endiol   C(=C(C(=CC(=C)O)O)O)O
5   C(=O)C(C(C(C(=C)O)O)O)O   R1_1_6_endiol   C(=C(C(C(C(=C)O)O)O)O)O
      C1=CN=C(C(C)O)O1         R1_4_2_strecker   C(=CN)OC(=O)C(C)O

```

The full graph is reconstructed by linking products to substrates and chaining through the triplets. Examples of two relatively short but different routes to dimethyl pyrazine are given below:

```

15   <Start> C(O)C(O)C(O)C(O)C(O)C=O R1_12_3_sugar C(O)C(O)C(O)C(=O)C(=O)C
      R1_2_1_retroaldol C(O)C(=O)C(=O)C R1_2_2_retroaldol C=O R2_5_4a_pyrazine
      CC-1NC(C)-CNC-1
      <Start> NC(C(O)C)C(=O)O R2_4_1_strecker CC(C=O)N R2_5_1_pyrazine
      CC1=NC(C)C=NC1 R1_5_3_pyrazine_oxidation CC-1NC(C)-CNC-1

```

20 The size of the soup, typically 100-1000 molecules, is determined at the start, and is limited only by computer memory considerations. At the start of a run this will be composed of starting components, which, in the case of the reaction to be simulated being a Maillard-type reaction amino acids and sugars only, e.g. for glucose and threonine (coded in SMILES):

```

25   "C(O)C(O)C(O)C(O)C(O)C=O
      C(O)C(O)C(O)C(O)C(O)C=O
      ...
      NC(C(C)O)C(=O)O
      NC(C(C)O)C(=O)O
30   ....."
```

There are duplicates of molecules, as the relative number of times a molecule appears simulates the concentration of that molecule in the soup. During, and at the end of a run, the soup will contain a list of end products that is the result of simulating the reactions many thousands of times. It also may contain duplicates, to simulate the
 5 relative concentration of end products, e.g.:

```

      "C(=O)(C(=O)C)O
      C(=O)(C(O)C(=O)C)O
      C(=O)(C(O)C(=O)C)O
10    C(=O)(C(O)C)O
      .....
      "
```

Central to the working of the program is a computer simulation of the chemical
 15 reactions (i.e. transformations) which actually may take place during the chemical process or reaction to be simulated. Each virtual reaction or transformation is coded as a programme function that conducts the following steps:

1. 2-D pattern match on substrate (input) molecule(s) according to the virtual reaction
- 20 2. Break bonds
3. Change atom hybridisation
4. Change bond types
5. Add bonds
6. Output product molecule(s)

25

In principle, this may be coded in any suitable computer-readable format, for example in SPL (Sybyl Programming Language^[3]) or any equivalent way. Such a programme may require a coding of the molecules and transformations or computer operations, which can be done e.g. in SMILES^[6] or SLN (the line notation from Tripos^[3] which is
 30 better compatible with SPL), which are then applied in the code for the Reaction Set.

The pattern matching step allows for fragment matching on the connection table of the reactive fragment necessary for the reaction to take place. Thus the chemical process is coded as a set of generic reactions which can act on a range of (different) starting molecules.

5

The IRG iterates through the Reaction Set, selecting reactions from the list of reactions and molecules from the 'soup' that relate to that reaction. Optionally, a 'filter' or selection criterion is build in, depending upon the specific case, which may e.g. help preventing polymerisation or will stop the simulation when desired compounds are
10 formed, or a certain level of compound(s) is formed, or other. Such filter or selection criterion can be e.g. an upper mass limit, or a lower mass limit, or the appearance of certain specific molecule or a group of molecules, molecular mass in some range, particular functionality of a compound, toxicity, etc.

15 The theory for kinetics for a simple chemical reaction: $A + B \rightarrow P$, where A and B are substrates and P is the product molecule is:

$$\frac{d[P]}{dt} = \frac{-d[A]}{dt} = k_{ABP} \cdot [A] \cdot [B]$$

20 where k_{ABP} is the rate constant for that reaction. It is in principle possible, but very time consuming, to calculate the rates of chemical reactions in solution or in an enzymatic environment from the free energy profile along the reaction coordinate. The free energy of activation has a simple relation to the rate constant in the transition state approximation:

$$k_{ABP} = \frac{k_B T}{h} e^{-\frac{\Delta G^\ddagger}{RT}}$$

25

Where

k_B = Boltzmann constant

T = temperature

h = Planck's constant
 ΔG^\ddagger = free energy of activation
 R = gas constant

- 5 ΔG^\ddagger consists of two components, the intrinsic part and the difference in free energy of solvation between the transition state and the reactants. The first can be calculated by either ab-initio or semi-empirical molecular orbital methods for both the transition state and the reactants. The difference in the free energies of solvation can be estimated using discrete solvent molecules or by continuum models. Simulation of energetic
- 10 details of the reaction, however, would require the search for transition states and their respective energetic minima. This would be an impossible task to do in a definite timescale given the present computing power. Therefore, in the present invention, it was decided that the simulation of the actual reaction steps together with their respective probabilities becomes the preferred option. As a result a 'reaction
- 15 probability' route approach has been adopted, using best guesses initially and preferably refining these empirically and/or by optimisation methods.

Discretising equation (1) the following is obtained:

$$\Delta[A] = -k_{ABP} \cdot [A] \cdot [B] \cdot \Delta t$$

20

Losing the time step Δt in the constant of proportionality, and describing values as probabilities, this may be written as:

$$\Delta(n(A)) \propto -p(R_{ABP}) \cdot p(A) \cdot p(B)$$

25 where

$n(A)$ = number of molecules of A in the Soup
 $p(R_{ABP})$ = relative 'probability' of Reaction A + B \rightarrow P
 $p(X)$ = probability of selecting molecule X from the Soup

The joint probability $p(A).p(B)$ may be simulated by randomly picking a pair of molecules {<molecule1>, <molecule2>}. This selection is biased by the 'concentrations' of molecule1 and molecule2 in the soup and therefore, over
5 successive selections, is a reasonable approximation to the probability. $p(R_{ABP})$ may be simulated by assigning a 'probability of reacting' to each reaction R, and randomly selecting the reactions. If the selected molecules match the requirements of the reaction R then they react and the products are added to the soup. In essence this is simulating that if A & B come into contact in the 'soup': if they can react they should do
10 so biased by some likelihood.

To facilitate scale-up and reduce computation time the reaction database (which is part of the reaction set) is preferably split into blocks, so that only selected reactions will occur within each block. The output from each block of reactions serves as input to one
15 or more further blocks.

This is structured in fig. 4 (wherein the reaction taken is a Maillard-type reaction, for illustration) according to the order in which reactions occur in the Maillard process. This refinement is not as strongly sequential as it may appear: parallel reactions may take
20 place within each block; the same reaction may occur in more than one block; and there is a high level of traffic between the blocks.

Alternatively to simulation of the reactions, estimations for determining one or more of the N processing parameters (and/or the reactant(s)) the simulation of complex
25 chemical reactions as set out herein before are derivable from a relationship between:

- composition analyses of compounds produced,
- processing parameters used for obtaining the composition analysis,
- reactants,

said composition analyses being an actual mass distribution obtainable from
30 performing at least 100 (preferably at least 1000) reactions involving heating reactants under predetermined and known processing parameters, analysing the reaction product obtained from each of the reactions above to provide composition analyses

thereof, encoding it as a mass distribution. In order to achieve this, samples may be produced under well defined standard conditions. The actual mass distribution may be obtainable by conventional chemical analysis of the reaction products or the volatile fraction thereof, such as GC and/or MS techniques. If so desired, this may be
5 combined by computerised processing of the analytical data. Needless to say, in view of the large number of experiments to be carried out, this (conducting the experiments and analysis) is preferably carried out in a robotised or automated way.

As an example, in the case of a Maillard-type reaction to be simulated, in brief, a
10 mixture of amino acid(s) and sugar(s) may be heated in solvent, cooled, and then extracted. The composition of volatile products may be determined by Gas Chromatography or similar separation technique. The identity of each peak may be determined by Mass Spectrometry from comparison with the generated fragmentation pattern of a library. From this a Molecular Mass Distribution (MMD) pattern can be
15 reconstructed, representing the frequency of masses of the product composition of each individual experiment. The final output of the computational IRG contains the 'soup' of molecules at the end of the run. This may be represented as a "Virtual Mass Distribution" (VMD) by taking relative frequencies binned by molecular weight. The experimental MMD may then be compared with the VMD.

20

Comparison of the experimental (= actual) mass distribution with the virtual mass distribution, as generated using IRG, yields information that can be used to update the IRG and/or reaction set. E.g., compounds which show up in the experimental results but are missing in the IRG results might implicate that an
25 elementary transformation is missing in the reaction database. Compounds present in the IRG results which are missing in the experimental mass distribution may originate from a probability of a certain transformation which is too high. The information thus acquired combined with the chemical knowledge of the user can be used to add or remove transformation steps and/or to change the probabilities of some of the
30 transformations, as is schematically given in figure 2.

The results described above, along with the full listing of the reaction paths, may be used as a guide to identifying where the output of the IRG may be improved by updating the values of the reaction rate parameters. The effect of such updates may easily be evaluated by running the updated IRG and comparing the results with the experimental data. If this results in an improvement the update is accepted, otherwise other updates are attempted.

The invention further relates to a computerized system comprising means for entering GC ('fingerprint') data and process variables to be set at the start of a chain of reactions and optional further data, and a computer programme to relate these. From such a relationship it is possible to predict process variables to obtain new desired fingerprint data, based upon already entered sensorical data, fingerprint data and process variables, and means for providing output.

In a preferred embodiment, the comparison or relationship between composition analyses of produced compounds in the form of actual and/or virtual mass distributions, and processing parameters used for obtaining the composition analysis and optional further data are obtainable using statistical methods. An example of such statistical methods may be a relationship method like linear- or non-linear regression, PLS, neural networks, gaussian procedures, etcetera.

The reaction rate parameters (probabilities) may be optimised by any suitable method. For example, the method as described below may be used.

In the case important process conditions are pH, T and S an objective or cost function related to the experimental measures is defined as:

$$\text{Error}(R(\text{pH}, T), S) = \text{false_positives}(S, \text{pH}, T) + \text{false_negatives}(S, \text{pH}, T);$$

where

R= the set of transformation rate parameters (i.e. probabilities) at the specified pH [high, med or low] and T (temperature of soup)

S = the start soup

false_positives = the number of molecules the IRG has incorrectly identified as being present in the final soup

false_negative = the number of molecules the IRG has failed to identify as being present in the final soup

Note that this does not take into account the peak height, but only the presence or absence of particular molecules. Then an objective function summed over the start soups for which there is experimental data may be defined:

10

$$O(R(pH, T)) = \sum_S \text{Error}(R(pH, T), S)$$

Clearly as $O(R(pH, T))$ approaches 0, the IRG is producing results closer to the experimental values. Defining the optimisation problem to be to optimise $R(pH, T)$, i.e.

15 the rate parameters for a given pH and temperature, such that $O(R(pH, T))$ is minimised. This is computationally expensive but may be achieved using a standard optimisation algorithm such as Sequential Quadratic Programming or a Genetic Algorithm. For other process variables that pH and T this works similarly.

20 Comparing the virtual mass distribution with the actual molecular mass distribution may be further supplemented with analysis of and comparison with e.g. sensory data or other data. Such sensory data may be obtained from analysing (e.g. using a sensory panel) the reaction products of the actual experiments, and preferably the volatile fraction thereof. The analysis of sensory data may involve statistical methods for mapping the sensory data. If sufficient data are then obtained, mathematical relationships between sensorical data and processing variables may then be derived.

30 References

[1] Molecular Simulations Inc., 9685 Scranton Road, San Diego, CA 92121-3752, USA.

- [2] Oxford Molecular Group PLC, The Medawar Centre, Oxford Science Park, Oxford OX4, 4GA, United Kingdom.
- [3] Tripos Inc., 1699 South Hanley Road, St. Louis, MO 63144, USA.
- [4] Vernin, G.; Parkyani C.; Barone R.; Chanon M.; Metzger J.; Computer Assisted Organic-Synthesis of Volatile Heterocyclic Compounds in Food Flavours, *Journal of Agriculture and Food Chemistry*, 1987, 35, 5, 761-768.
- [5] Azario, P.; Arbelot M.; Baldy A.; Microcomputer Assisted Retrosynthesis (MARS), *New Journal of Chemistry*, 1990, 14, 12, 951-956.
- [6] Bador, P. et al; Les Systemes Informatiques de Recherche d'Information sur les Reactions Chimiques et les Systemes de Synthese Assistee par Ordinateur, *New Journal of Chemistry*, 1992, 16, 3, 413-423.
- [7] Prickett, S.E.; Mavrovouniotis, M.L.; Construction of Complex Reaction Systems – II Molecule Manipulation and reaction application algorithms, *Computers Chem. Engng.*, 1997, 21, 11, pp 1237-1254
- [8] Weininger, D.; SMILES, a chemical language and information system, *Journal of Chemical Information and Computer Science*, 1998, 28, 1, 31-36
- [9] Lohn, J.D.; Evolving Catalytic Reaction Sets using Genetic Algorithms, IEEE World Congress on Computational Intelligence, Anchorage, Alaska. 1998, 87-492
- [10] Schuster, P.; Dynamical Systems and Cellular Automata, J. Demongeot et al. (Eds), *Academic Press*, 1985, 255-267
- [11] Banzhaf, W. et al; Emergent Computation by Catalytic Reactions, *Nanotechnology*, 1996, 7, 307-314
- [12] Kauffman, S.A.; The Origins of Order, *Oxford University Press*. 1993, 303-305

25

Examples

Example 1

In figure 3, an example is given how an assembly of actual and virtual experimentation, and sensory analysis may be used jointly.

30

Example 2.

This example gives a high level pseudocode for how the IRG may be coded.

```

Initialise Soup, Reaction Set
  Loop
    Loop through Reaction Blocks*
    Select Random reaction
    If (transformation probability > random number)
      Select random reactant(s)
      If reactant(s) are correct for reaction
        Remove bonds
        Change atom type & hybridisation
        Add bonds
        If (mass of product < mass limit)**
          Remove reactants from Soup
          Add product(s) to Soup
        Endif
      Endif
    Endif
  Endloop
Endloop

```

5

Italics indicate optional computer instructions:

* if reaction blocks are used

** if a mass limit is used

10

Example 3.

This example gives the SPL code for the main body of the IRG, similar to Example 2

```

=====

```



```
uims define expression_generator iterate yes

    setvar fh %open($filename3)
5   setvar fh2 %open($filename5)
    %write($fh2 Time $chkprod)

    # Call blocks of reactions.

10  FOR blocks in %range(1 $blocknum 1)

    %write($fh "    ")
    %write($fh "Block" $blocks)
    %write($fh "    ")
15  %write($fh2 "    ")
    %write($fh2 "Block" $blocks)
    %write($fh2 "    ")

    setvar inns %set_unpack($inputset[$blocks])
20
    FOR those in $inns
        setvar soupmix[$blocks] $soupmix[$blocks] $soupmix[$those]
    ENDFOR

25  # iterate on soupmix[$blocks]

    FOR backups in %range(1 10 1)
        FOR u in %range(1 10 1)
            setvar v 0
30        FOR t in %range(1 %math($icycles / 100) 1)
            setvar randomnu %math($lastprob[$blocks] * %rand())
            setvar reactionnumber ""
```

```
FOR roulette in %range($totalnum[$blocks] 1 -1)
  IF %LTEQ($randomnu $cumulist[$blocks][$roulette])
    setvar reactionnumber $roulette
  ENDIF
5  ENDFOR
  setvar runreaction %arg($reactionnumber $tallist[$blocks])
  setvar reacttype %substr($runreaction 1 2)

  IF %streql(R1 $reacttype)
10  # Call unimolecular reaction with random reactants

    FOR alpha in %range(1 4 1)
      setvar soupsize %count($soupmix[$blocks])
      setvar j %math(%int(%math(%math($soupsize - 0.0002) * %rand())) +
15      1.0001)

      setvar soupmol %arg($j $soupmix[$blocks])
      IF %gt(%strlen($soupmol) 0)
        setvar scommand %cat('%' $runreaction '(' $soupmol ')')
        setvar mproduct %eval($scommand)
20      IF %gt(%strlen($mproduct) 1)
        setvar soupmix[$blocks] %item_remove($j $soupmix[$blocks])
        setvar mproduct %remwater("$mproduct")
        setvar soupmix[$blocks] $soupmix[$blocks] $mproduct
        %uppaths($soupmol $runreaction "$mproduct" )
        %uptable($soupmol $runreaction "$mproduct" )
        %upretable($runreaction)
        setvar v %math($v + 1)
      ELSE
30      ENDIF
    ENDIF
  ENDIF
ENDFOR
```

ELSE

5 # Call bimolecular reaction with random selections of two reactants

```
    IF %streq(R2 $reacttype)
      FOR alpha in %range(1 4 1)
        setvar soupsize %count($soupmix[$blocks])
10      setvar n %math(%int(%math(%math($soupsize - 0.0002) * %rand())) +
        1.0001)

        setvar first %arg($n $soupmix[$blocks])
        setvar j %math(%int(%math(%math($soupsize - 0.0002) * %rand())) +
        1.0001)
15      IF %eq($j $n)
        ELSE
          setvar second %arg($j $soupmix[$blocks])
          IF %gt(%strlen($first) 0)
            IF %gt(%strlen($second) 0)
20            setvar soupmoles %cat($first . $second)
            setvar scommand %cat('%' $runreaction '(' $soupmoles "'")
            setvar mproduct %eval($scommand)
            IF %gt(%strlen($mproduct) 1)
              IF %gt($n $j)
25              setvar soupmix[$blocks] %item_remove($n
              $soupmix[$blocks])

              setvar soupmix[$blocks] %item_remove($j
              $soupmix[$blocks])

              ELSE
30              setvar soupmix[$blocks] %item_remove($j
              $soupmix[$blocks])
```

```

                                setvar soupmix[$blocks] %item_remove($n
$oupmix[$blocks])
                                ENDIF
                                setvar mproduct %remwater("$mproduct")
5                                setvar soupmix[$blocks] $oupmix[$blocks] $mproduct
                                %uppaths($first $runreaction "$mproduct" )
                                %uptable($first $runreaction "$mproduct" )
                                %uppaths($second $runreaction "$mproduct" )
                                %uptable($second $runreaction "$mproduct" )
10                                %upretable($runreaction)
                                setvar v %math($v + 1)
                                ELSE
                                ENDIF
                                ENDIF
15                                ENDIF
                                ENDIF
                                ENDFOR
                                ENDIF
                                ENDIF
20                                ENDFOR

                                setvar chksum ""

                                # check for the presence of compounds in current soupmix.
25                                IF %streql(yes $pcheck)
                                    FOR x in %range(1 %count($oupmix[$blocks]))
                                        setvar dummy %smiles_to_mol(m1 %arg($x $oupmix[$blocks]))
                                        FOR y in %range(1 %count($chkprod))
30                                        IF %sln_search2d(m1 %arg($y $chkprod) mutual norm 1)
                                            IF $chksum[$y]
                                                setvar chksum[$y] %math(1 + $chksum[$y])
                                            
```

20

```

        ELSE
            setvar chksum[$y] 1
        ENDIF
    ENDIF
5    ENDFOR
    ENDFOR
    ENDIF
    %write($fh2 %arg(4 %time()) $chksum)
    %write($fh %arg(4 %time()) $v)
10    ENDFOR

    # Make a temporary save of the soupmix and paths

    echo "Saving backup file ..."
15    %tmp_file_save(%math($backups * 10) $blocks $backupname)
    echo "Backup file saved."

    ENDFOR

20    IF %streql(yes $timevms)

        # Write multiple virtual mass spec graph data to file
        # Uses the current block of the soupmix not rather than the whole.

25        setvar size 1
        setvar mass ""
        setvar w %printf("%02d" $blocks)
        setvar fh3 %open(%cat($vmsname $w .txt))

30        FOR j in %range(%count($soupmix[$blocks]) 1 -1)
            setvar dummy %smiles_to_mol(m1 %arg($j $soupmix[$blocks]))
            setvar mass[$j] %int(%molmass(m1))
        ENDFOR
    ENDIF
END
```

```

ENDFOR

setvar mass %sortn($mass)
setvar n 1
5
FOR k in %range(%math(%count($mass) - 1) 1 -1)
    IF %eq(%arg($k $mass) %arg(%math($k + 1) $mass))
        setvar n %math($n + 1)
        setvar $mass %item_remove(%math($k + 1) $mass)
10    ELSE
        %write($fh3 %arg(%math($k + 1) $mass) %math($n * $size))
        setvar n 1
    ENDIF
ENDFOR
15    %write($fh3 %arg(1 $mass) %math($n * $size))
    %close($fh3)
ENDIF
ENDFOR
%close($fh2)
20 %close($fh)

```

Example 4

Basic rules for writing each reaction in SMILES notation and three examples of
25 reactions typical for Maillard, as found in literature and how they are coded into
SMILES strings and reactions for the IRG.

Basic rules for SMILES:

Instructions for adding to data base:

30 # Is this an UNARY or a BINARY reaction type?

UNARY

R1_1_1_sugar

Pattern for matching against, atoms start counting at 0 from the left
Binary reactions have two patterns, atom numbers continue from the first pattern
onto the second
C(=O)C(O)C(O)
5 # The numbers of atoms which have restrictions to the atoms joined to them
-1 terminates the list
0 3 4 5 -1

These are the restrictions as atom type letter and hybridisation number
10 # H3 H3 H3C3 H3
Other restriction state if at least one Hydrogen must be present
N N Y N
Catstring is for adding water if required, the number assigned to it
follows on from the last atom of the pattern
15 # Both unary and binary reactions use this. If not used then NA replaces it.
NA

bonds to be removed as the numbers of the atoms which are on each end
2
20 # 2 3
4 5

bonds to be added as the numbers of the atoms on each end with bondtypes
1
25 # 2 3 2

Note: The numbering in each of the 2D representations is the same as that used
for the atoms on converting into SMILES notation.

30 # Example 4a: R2_3_15_1_pyrroline

23



reaction in SMILES code:

BINARY

5 R2_3_15_1_pyrroline

OC(=O)C1CCCN1C(=O)C(=O)C

0 3 4 5 6 7 8 12 -1

H3 H3 H3 H3 H3 H3 H3 H3

10 N N N N N N N N

NA

4

0 1 13 37 89

3

15 0 12

3 7 2

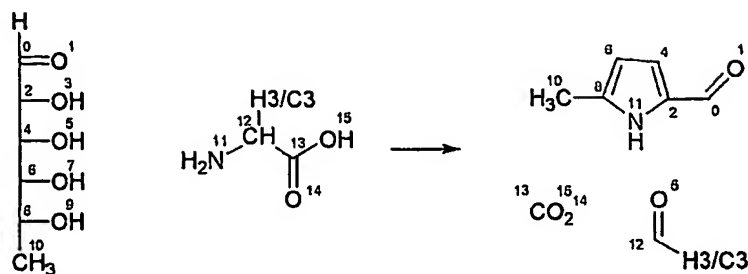
8 9 1

Added 27.4.99 (SR)

J.E. Hodge, F.D. Mills and B.E. Fisher, Cereal Sci. Today 17, 34-40 (1972)

20 # Checked 10.5.99 (FH)

Example 4b: R2_10_1b_rS+AAMeCHOpyrrol



BINARY

R2_10_1b_rS+AAMeCHOpyrrol

5 C(=O)C(O)C(O)C(O)C(O)C

NCC(=O)O

0 3 5 7 9 10 11 12 15 -1

H3 H3 H3 H3 H3 H3 H3 H3 H3C3 H3

N N N N N N N Y N

10 NA

9

2 3 2 4 4 5 6 7 6 8 8 9 11 12 12 13 13 15

6

2 4 2

15 2 11 1

6 8 2

8 11 1

12 5 2

13 15 2

20 # water molecules not explicitly drawn

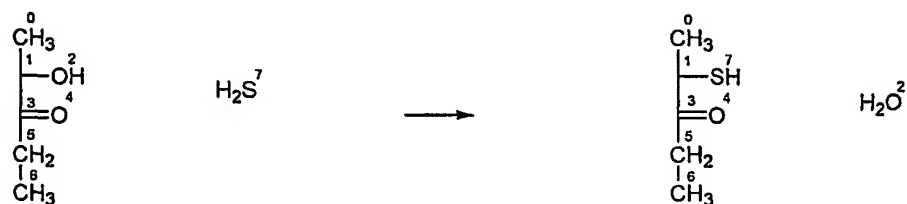
Added 20.9.99 (SR). Comparable to R2_10_1b_asugarAA but on rhamnose.

R.Tressl, E. Kersten, C. Nittka and D. Rewicki. Maillard Reactions

in Food and Health, Proceedings of 5th Int. Symp. on Maillard Reactions

26 aug - 1 sept 1993. (RSC Special publication 151, 1994, p.51)

25

Example 4c: R2_8_14b_2thiopent3on

5 BINARY

R2_8_14b_2thiopent3on

CC(O)C(=O)CC

S

0 1 2 5 6 7 -1

10 H3 H3 H3 H3 H3 H3

N N N N N N

NA

1

1 2

15 1

1 7 1

Added 17.8.99 (FH)

changed to OH/SH-substitution J.Agric.Food Chem.1999,47,1626. - 25.8.99 (FH)

20

Example 5

Example of blocks of reactions as may be used in the reaction database, according to the order in which reactions occur in the Maillard process, but the same reaction may occur in more than one block (figure 4). Other arrangements are possible.

25

Example 6.

Experimental validation with virtual mass distribution (VMD) was obtained by comparison of an actual mass distribution (MMD) with a virtual mass distribution. The conditions for the simulations were: 100 molecules glucose, 100 molecules threonine, 6000 iterations, pH=7, Temperature=120° Celsius. The conditions
5 for the real experiment are: equimolar mixture of glucose and threonine, in a buffered solution pH=7, processed during 1 hour at 120° Celsius.

In figure 5, the MMD, the VMD, and the matches have been printed in different fonts. Clearly, the formation of formic acid, acetic acid, glycolic aldehyde, hydroxyacetone,
10 lactones, oxazoles, and some pyrazines can be seen. There are also a number of mismatches: a number of start components and intermediates, such as threonine, formaldehyde, acetaldehyde, and various sugar derivatives are present in the IRG 'soup' but not in the experimental results. The IRG has also failed to match some the substituted pyrazines as well as some of the smaller peaks.

15

Claims

1. Method for simulating a chemical process, which process may comprise multiple branches of reaction pathways and/or feed back/forward loops and/or parallel reaction branches by an iterative procedure of applying:
 - a 'Reaction Set' describing transformations that may take place in the chemical process that is to be simulated, and probabilities of said transformations
 - a 'Soup' of molecules representing the state of the system.
2. Method according to claim 1, wherein during the iterative procedure part or all of the reaction products are added back to the Soup.
3. Method according to claim 1-2, wherein the Soup at the start of the reaction is equal to the starting mixture of molecules.
4. Method according to claim 1-3, wherein the 'Reaction Set' comprises:
 - a reaction database, comprising various transformations that may take place in the chemical process to be simulated,
 - a reaction kinetic database, comprising relative probabilities for the transformations in the reaction database.
5. Method according to claim 1-4, wherein iterative procedure is a computer-readable format encoded by:
 - Initialise Soup and Reaction Set (containing reaction database and reaction kinetic database) and optionally Filter
 - Loop
 - Loop through reaction blocks*
 - Select Random reaction
 - If (transformation probability > random number)
 - Select random reactant(s)
 - If reactant(s) are correct for reaction

```
Remove bonds
Change atom type & hybridisation
Add bonds
If (reaction product equals Filter)
    Remove reactants from Soup
    Add product(s) to Soup
Endif
Endif
Endif
Endloop
Endloop
```

or any functional equivalent thereof, wherein the Italics indicate optional computer instructions.

6. Method according to claim 1-5, wherein wherein the iterative procedure is coded as a computer programme directly loadable in the internal memory of a computer.
7. Process according to claim 1-6, wherein an actual mass distribution is obtained by performing part or all of the reactions that are simulated, wherein the actual mass distribution is compared with the Soup, and wherein the difference of the actual mass distribution and the Soup is used to update the Reaction Set.
8. Process according to claim 7, wherein the actual mass distribution is obtainable by conventional chemical analysis of the reaction products or the volatile fraction thereof.
9. Process according to claim 8, wherein the conventional chemical analysis involves Gas Chromatography and/or Mass Spectroscopy techniques.
10. Process according to claim 9, wherein the chemical analysis is combined by computerised processing of the analytical data.

11. Process according to claim 7-10, wherein the reactions performed to obtain the actual mass distribution data are carried out in a robotised way.
12. A computer program product directly loadable into the internal memory of a digital computer, comprising software code portions for the simulation of complex chemical reaction pathways by iteratively applying a set of operations to:
- a Soup of molecules representing the current state of the system,
 - a 'Reaction Set' describing transformations that may take place in the chemical process that is to be simulated, and probabilities of said transformations to yield molecules.
13. A computer programme product directly loadable into the internal memory of a digital computer, comprising software code portions coding for:
- Initialise Soup and Reaction Set (containing reaction database and reaction kinetic database)
- Loop
- Loop through reaction blocks*
- Select Random reaction
- If (transformation probability > random number)
- Select random reactant(s)
- If reactant(s) are correct for reaction
- Remove bonds
- Change atom type & hybridisation
- Add bonds
- If (reaction product equals Filter)*
- Remove reactants from Soup
- Add product(s) to Soup
- Endif*
- Endif
- Endif
- Endloop
- Endloop

or any functional equivalent thereof, wherein the *Italics* indicate optional computer instructions.

14. Computerized system comprising means for entering mass distribution data, process variables to be set at the start of a chain of reactions, reactants, and a computer programme for predicting process variables and/or reactants to obtain new desired mass distribution data using an iterative procedure, based upon already entered mass distribution data, process variables, and reactants and means for providing output.
15. Process according to claim 1, wherein the simulation is obtainable by iteratively applying a set of operations or computer instructions using a computer programme to:
 - A 'Soup' of molecules representing the current state of the system
 - A 'Reaction Set' describing transformations and probabilities that may take place in the chemical process to be simulated,to produce molecules, for simulating complex chemical reactions when such product is run on a computer, and wherein the iteration is effected by a computer programme directly loadable in the internal memory of a computer, and wherein the computer programme contains two main elements:
 - computer instructions for running the reactions using the Reaction Set,
 - computer instructions for the iterative procedure of running the reactions, selecting molecules, and producing output.
16. Process according to claim 15, wherein during the iterative procedure the newly formed compounds are added back to the Soup, and form (part of) the virtual mass distribution.
17. Process according to claim 15 or 16, wherein the Soup at the start of the reaction is equal to the starting mixture of molecules.

18. Computerized system comprising means for entering fingerprint data or reactants and process variables to be set at the start of a chain of reactions, and a computer programme for predicting process variables to obtain new desired fingerprint data using an iterative procedure, based upon already entered fingerprint data and process variables, and means for providing output.

1/4

Fig.1.

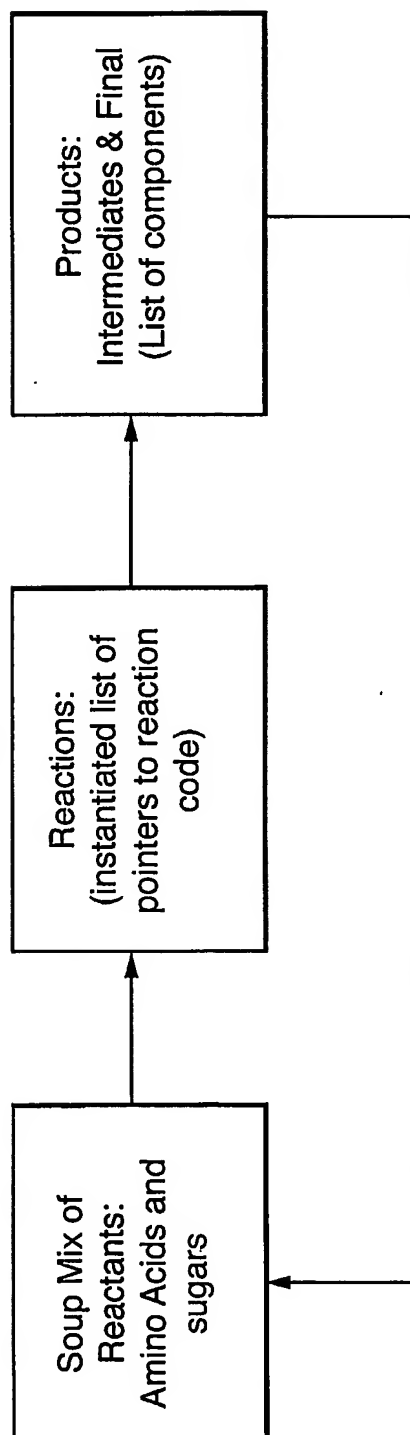
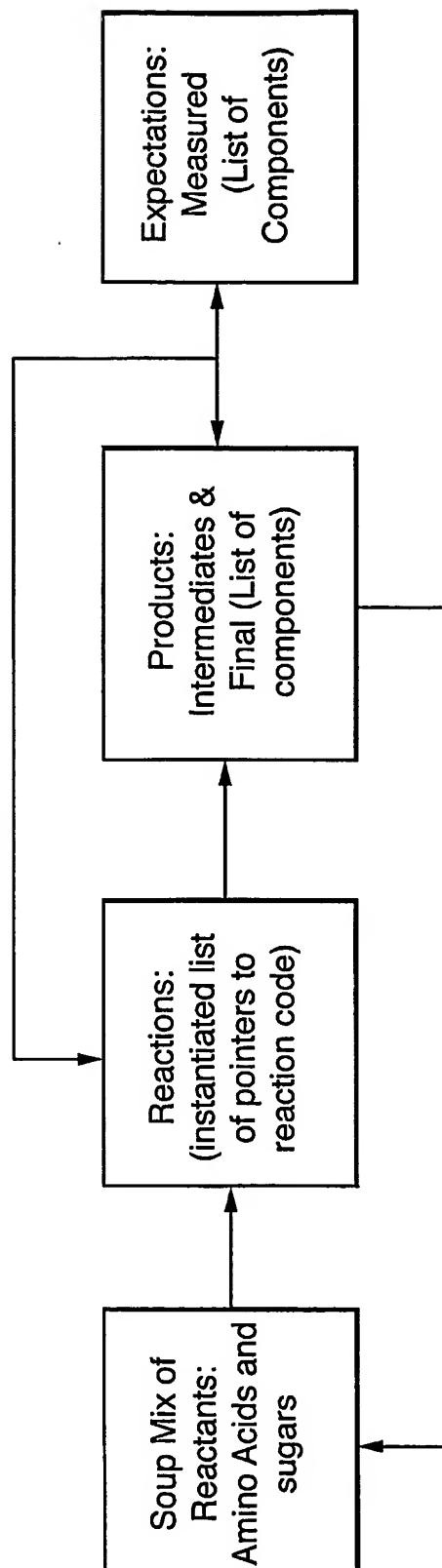
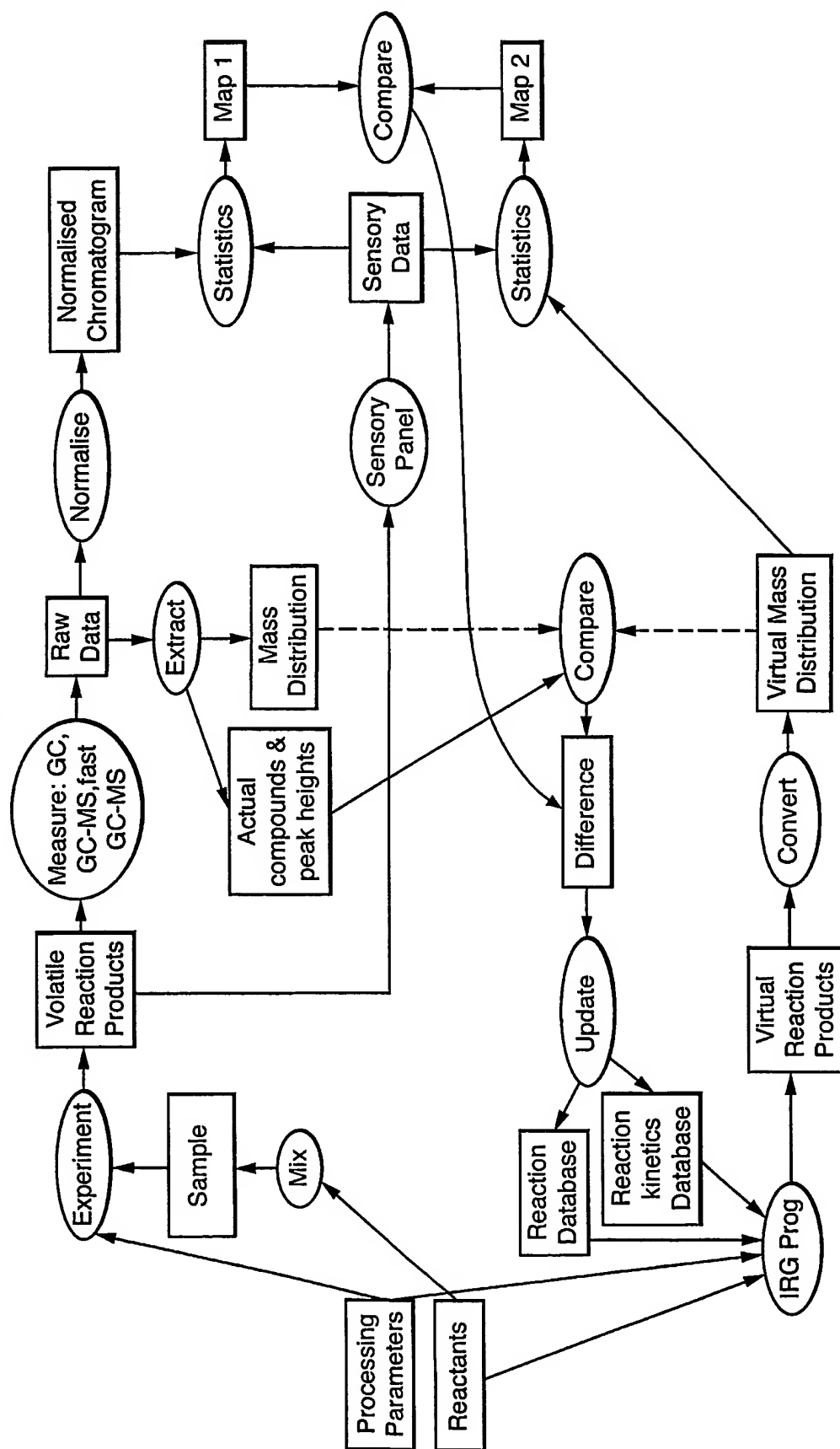


Fig.2.

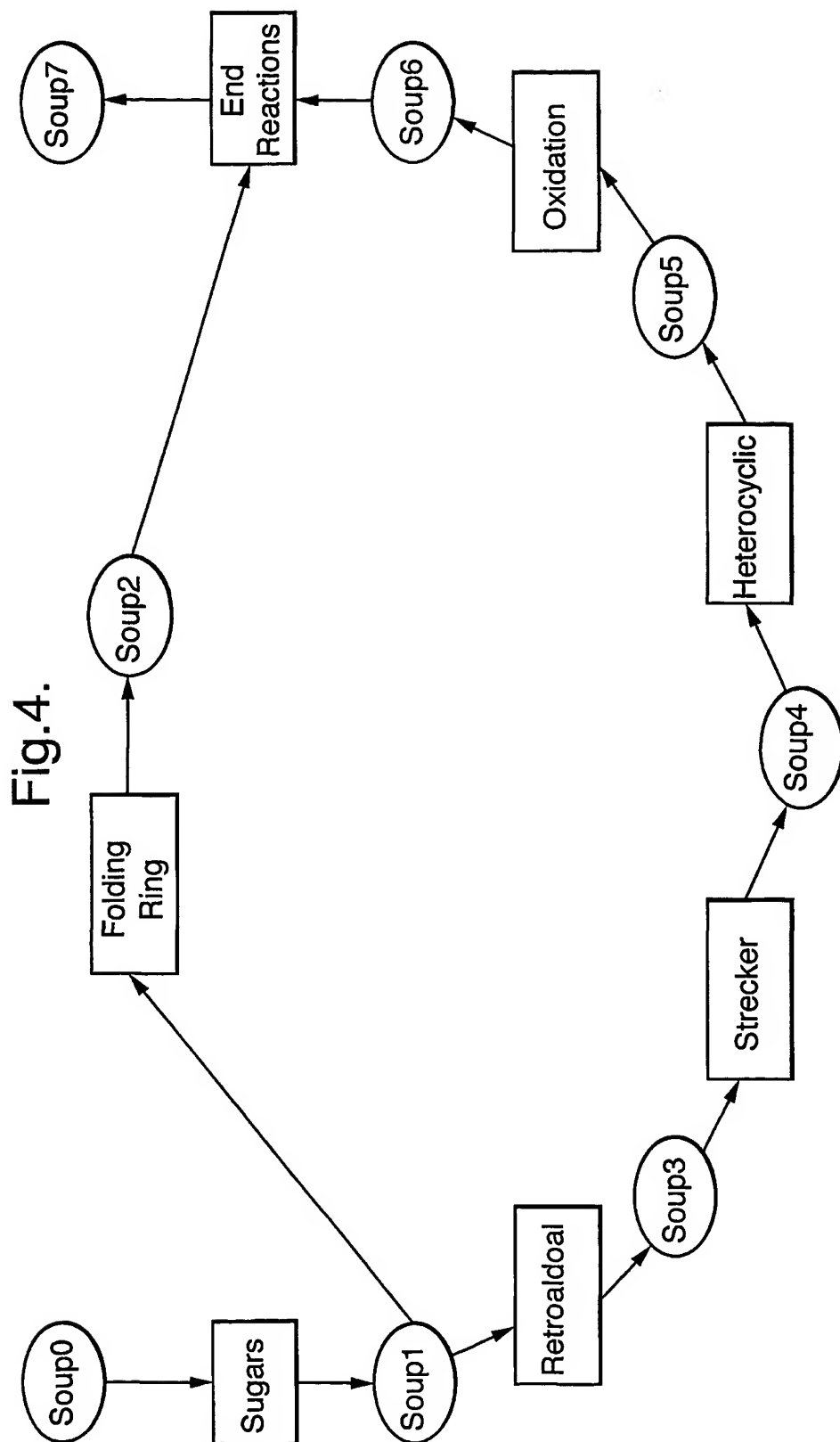


2/4

Fig.3.



3/4



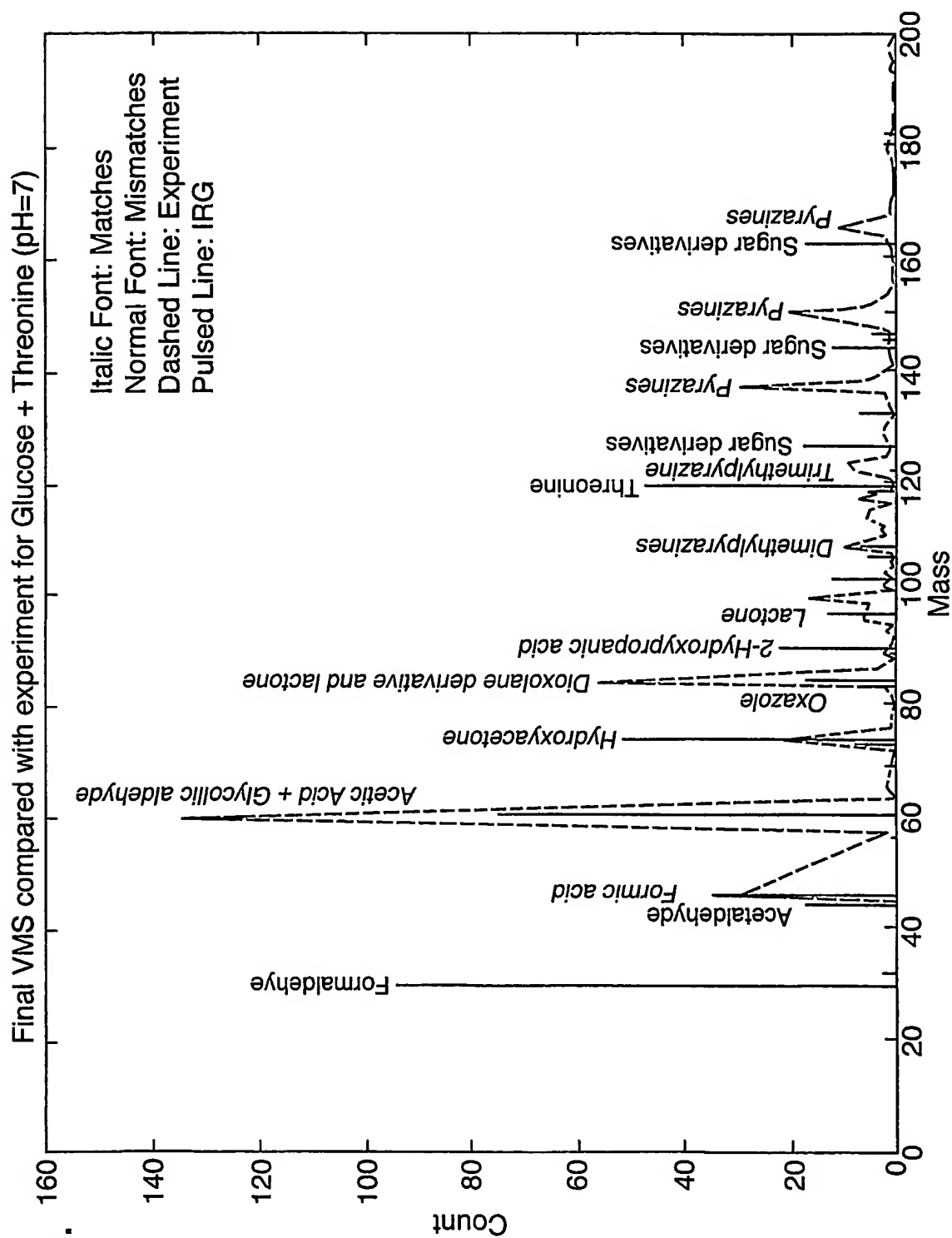


Fig.5.

INTERNATIONAL SEARCH REPORT

International Application No.

PCT/EP 01/07235

A. CLASSIFICATION OF SUBJECT MATTER
 IPC 7 G05B17/02

According to International Patent Classification (IPC) or to both national classification and IPC

B. FIELDS SEARCHED

Minimum documentation searched (classification system followed by classification symbols)

IPC 7 G05B

Documentation searched other than minimum documentation to the extent that such documents are included in the fields searched

Electronic data base consulted during the International search (name of data base and, where practical, search terms used)

EPO-Internal

C. DOCUMENTS CONSIDERED TO BE RELEVANT

Category *	Citation of document, with indication, where appropriate, of the relevant passages	Relevant to claim No.
A	<p>J. LOHN ET AL: "EVOLVING CATALYTIC REACTION SETS USING GENETIC ALGORITHMS" PROCEEDINGS OF THE 1998 IEEE WORLD CONGRESS ON COMPUTATIONAL INTELLIGENCE, 4 May 1998 (1998-05-04), pages 487-492, XP000938334 USA cited in the application page 487, left-hand column, line 1 -page 488, right-hand column, line 17</p> <p style="text-align: center;">--- -/--</p>	1,12,13



Further documents are listed in the continuation of box C.



Patent family members are listed in annex.

* Special categories of cited documents:

- *A* document defining the general state of the art which is not considered to be of particular relevance
- *E* earlier document but published on or after the international filing date
- *L* document which may throw doubts on priority claim(s) or which is cited to establish the publication date of another citation or other special reason (as specified)
- *O* document referring to an oral disclosure, use, exhibition or other means
- *P* document published prior to the international filing date but later than the priority date claimed

T later document published after the international filing date or priority date and not in conflict with the application but cited to understand the principle or theory underlying the invention

X document of particular relevance; the claimed invention cannot be considered novel or cannot be considered to involve an inventive step when the document is taken alone

Y document of particular relevance; the claimed invention cannot be considered to involve an inventive step when the document is combined with one or more other such documents, such combination being obvious to a person skilled in the art.

Z document member of the same patent family

Date of the actual completion of the international search

3 October 2001

Date of mailing of the international search report

15/10/2001

Name and mailing address of the ISA

European Patent Office, P.B. 5818 Patentlaan 2
 NL - 2280 HV Rijswijk
 Tel. (+31-70) 340-2040, Tx. 31 651 epo nl,
 Fax: (+31-70) 340-3016

Authorized officer

Kelperis, K

INTERNATIONAL SEARCH REPORT

International Application No

PCT/EP 01/07235

C.(Continuation) DOCUMENTS CONSIDERED TO BE RELEVANT

Category *	Citation of document, with indication, where appropriate, of the relevant passages	Relevant to claim No.
A	<p>R.MOROS ET AL: "A GENETIC ALGORITHM FOR GENERATING INITIAL PARAMETER ESTIMATIONS FOR KINETIC MODELS OF CATALYTIC PROCESSES" COMPUTERS AND CHEMICAL ENGINEERING, vol. 20, no. 10, October 1996 (1996-10), pages 1257-1270, XP000949232 UK page 1258, left-hand column, line 3 -page 1260, right-hand column, line 10 ---</p>	
A	<p>J.SRINIVASALU ET AL: "BROWNIAN DYNAMICS SIMULATIONS OF DIFFUSION CONTROLLED REACTIONS WITH FINITE REACTIVITY" JOURNAL OF CHEMICAL PHYSICS, vol. 107, no. 6, 8 August 1997 (1997-08-08), pages 1915-1921, XP000955643 USA page 1915, left-hand column, line 1 -page 1918, left-hand column, line 28 ---</p>	
A	<p>T.YUN PARK ET AL: "A HYBRID GENETIC ALGORITHM FOR THE ESTIMATION OF PARAMETERS IN DETAILED KINEMATIC MODELS" COMPUTERS AND CHEMICAL ENGINEERING, vol. 22, 1998, pages S103-S110, XP000955654 UK page S103, left-hand column, line 1 -page S106, left-hand column, line 9 ---</p>	
A	<p>WO 96 41822 A (SELLIERS JACQUES DE ;SOLVAY (BE)) 27 December 1996 (1996-12-27) page 19, line 33 -page 21, line 24 ---</p>	14,18
A	<p>US 6 056 781 A (MCCROSKEY PATRICK S ET AL) 2 May 2000 (2000-05-02) column 3, line 58 -column 16, line 7 ---</p>	14,18
A	<p>D.A.VOSS ET AL: "A LINEARLY IMPLICIT PREDICTOR-CORRECTOR METHOD FOR REACTION-DIFFUSION EQUATIONS" AN INTERNATIONAL JOURNAL :COMPUTERS AND MATHEMATICS WITH APPLICATIONS, vol. 38, no. 11-12, December 1999 (1999-12), pages 207-216, XP000987389 UK page 207, line 1 -page 211, line 21 -----</p>	14,18

INTERNATIONAL SEARCH REPORT

Information on patent family members

Application No

PCT/EP 01/07235

Patent document cited in search report		Publication date	Patent family member(s)	Publication date
WO 9641822	A	27-12-1996	BE 1009406 A3	04-03-1997
			AT 177761 T	15-04-1999
			AU 704563 B2	29-04-1999
			AU 6124696 A	09-01-1997
			BR 9609240 A	11-05-1999
			CA 2220316 A1	27-12-1996
			CN 1192222 A	02-09-1998
			CZ 9703985 A3	18-03-1998
			DE 69601794 D1	22-04-1999
			DE 69601794 T2	07-10-1999
			WO 9641822 A1	27-12-1996
			EP 0830394 A1	25-03-1998
			ES 2131945 T3	01-08-1999
			HU 9803021 A2	28-04-1999
			JP 2000500060 T	11-01-2000
			NO 975780 A	06-02-1998
			PL 323911 A1	27-04-1998
			US 6144897 A	07-11-2000
US 6056781	A	02-05-2000	US 5740033 A	14-04-1998